



Reducing HPC Network Requirements:

Exploiting overlap between communication and computation

Jose Carlos Sancho

jcsancho@lanl.gov

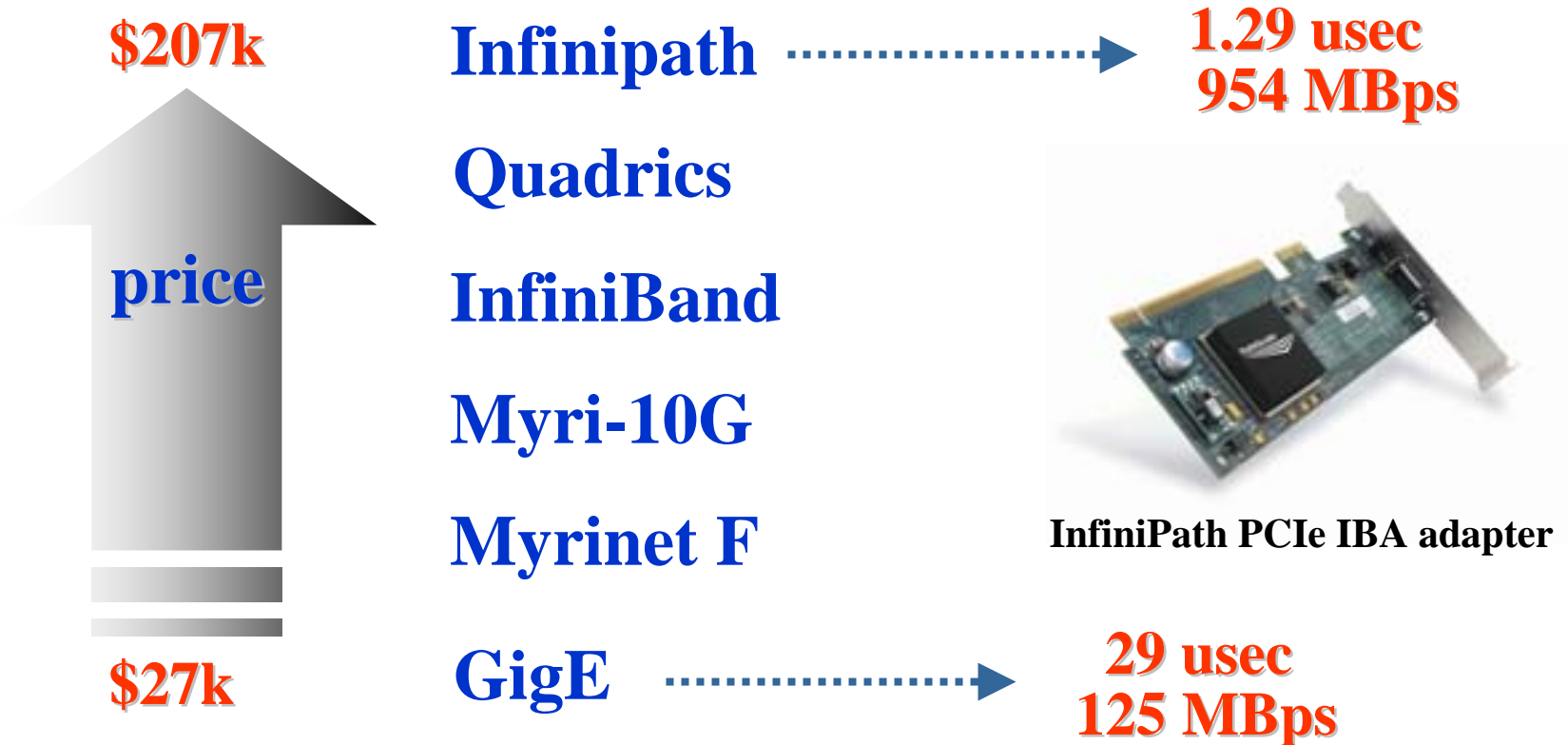
Darren J. Kerbyson, Kevin J. Barker, and Kei Davis

Performance and Architectures Lab (PAL)

- Why overlapping?
- Method to analyze the overlap
- Results on scientific applications
- Conclusions

Performance and cost of HPC networks

- Price is often related to performance



Prices approximated for a 128-node system
 Source: "Cluster Interconnects: The Whole Shebang", April 2006

Balancing a computer performance

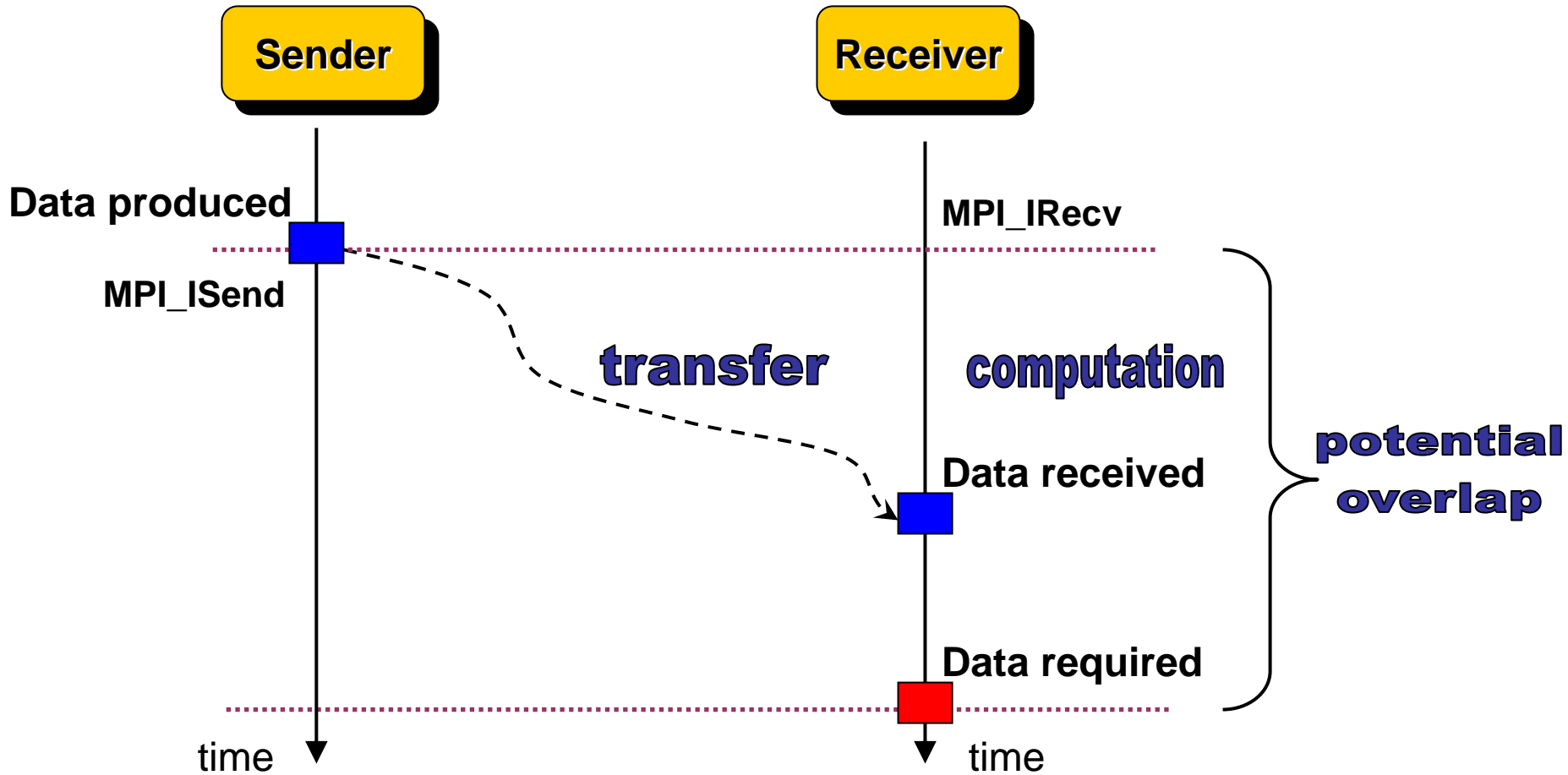
- Performance is strongly dependent on the applications
- Usage of application performance models under various network profiles to determine the network requirements
- Satisfying the needs for all applications would require extremely expensive networks

Network performance impact on LANL applications

| Latency | Bandwidth |
|---------|-----------|
| Sweep3D | SAGE |
| Partisn | |

High network bandwidth and low network latency are expensive !

Overlapping communication with computation



If potential overlap is large enough then network requirements can be lowered

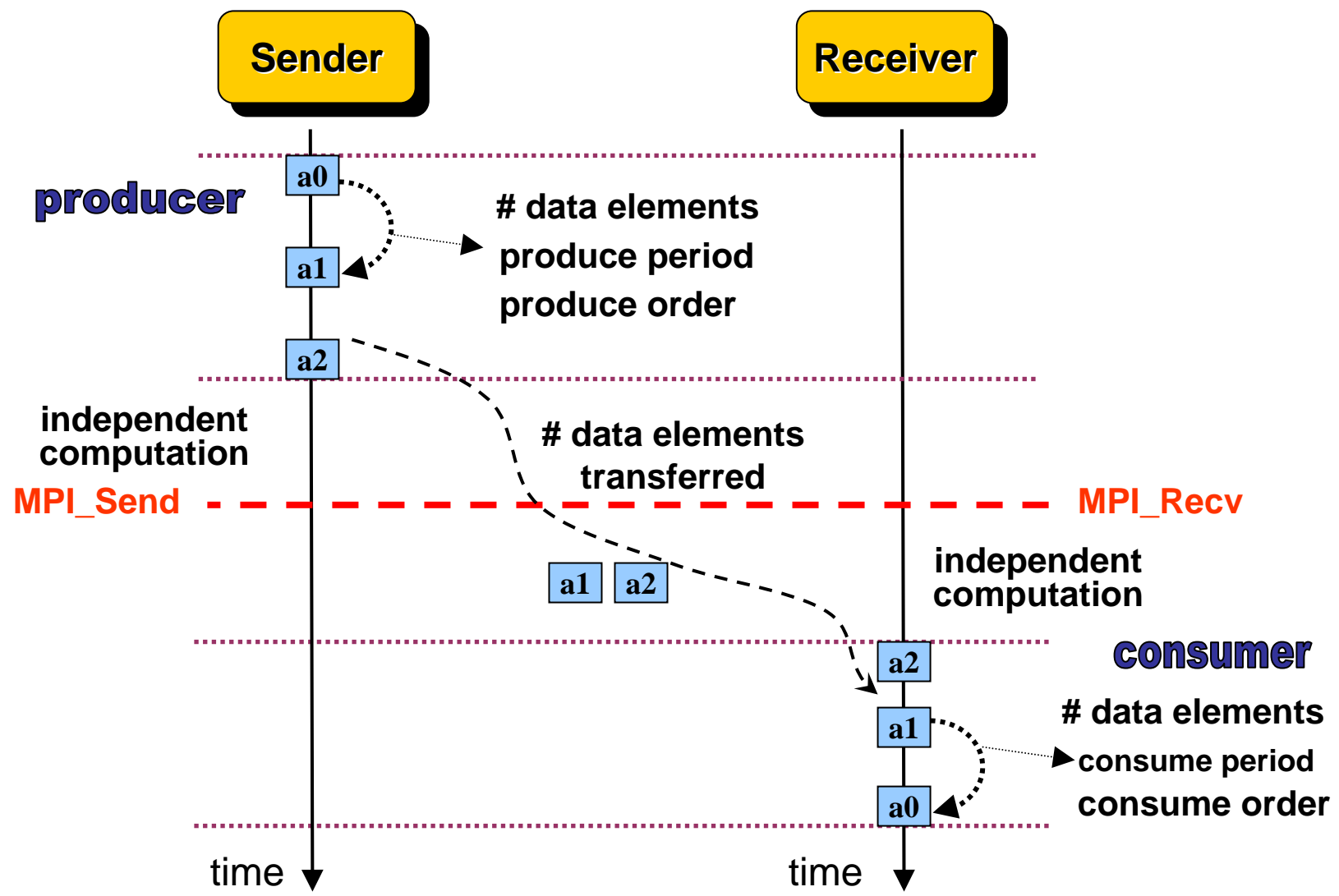
Overlapping in scientific applications

- Overlapping is rarely utilized and mostly used to hide some computation tasks on the communicator buffers
- Exploiting the overlap is not easy:
 - ◆ Rewriting the application communication subsystem and also some parts of the code
 - ◆ Ensuring the correctness of the application
- Lack of performance models to estimate the performance improvement of overlapping for an application of interest



**We need to have some tool to
assess the benefit of overlapping
in scientific applications**

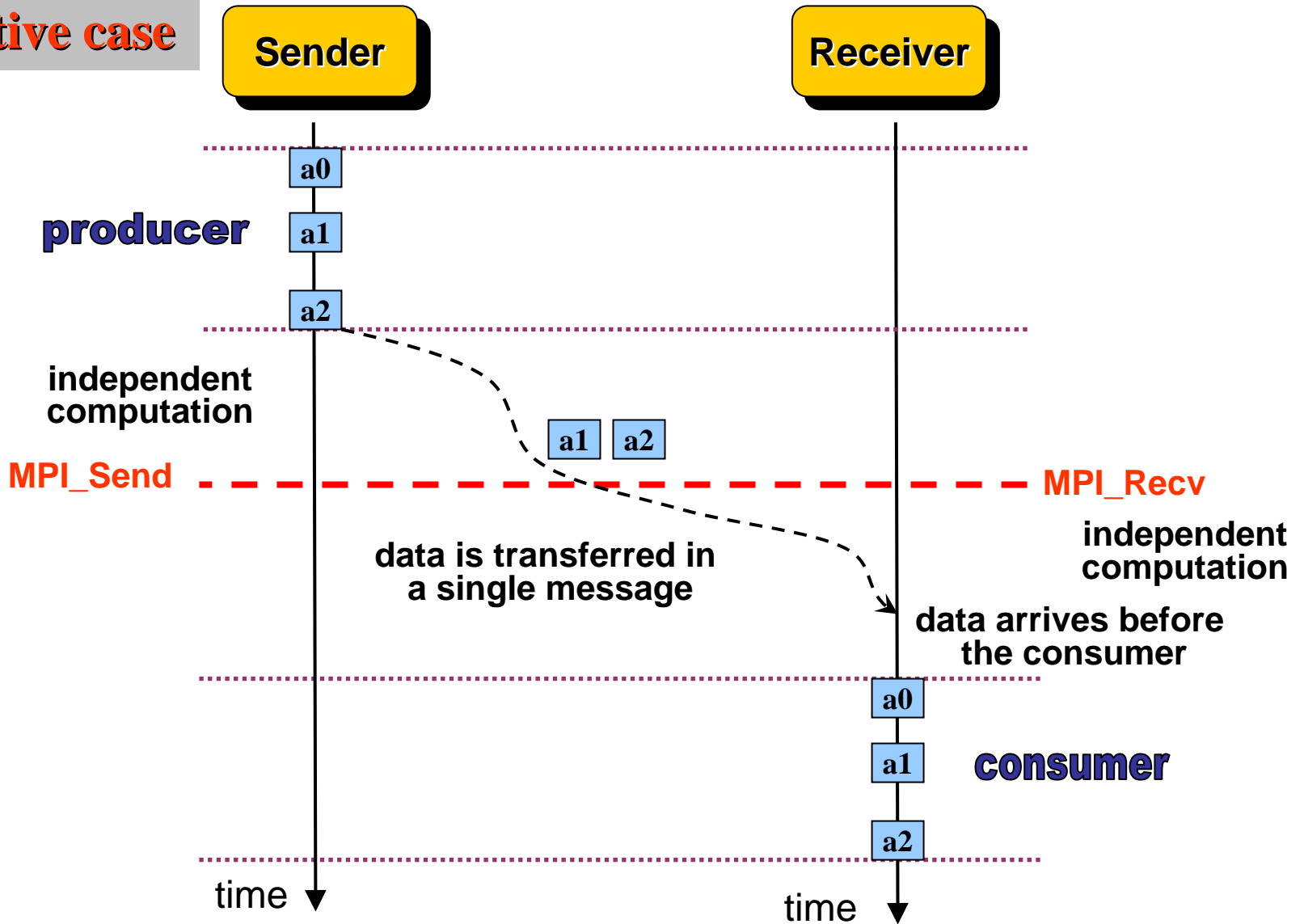
Illustrating the method on a point-to-point communication

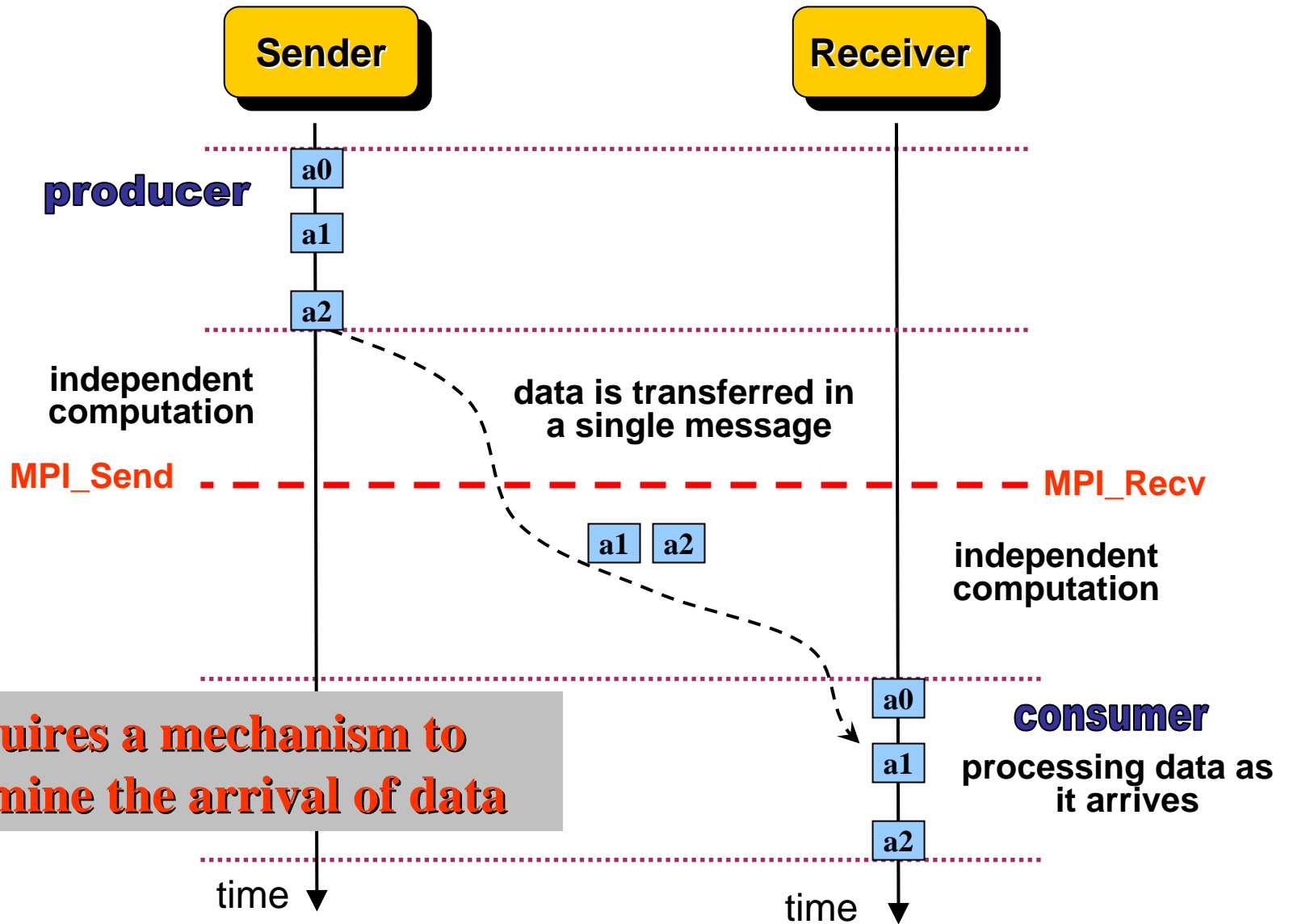


- 1) Independent work
- 2) Independent work plus the consumer dependent work
- 3) Independent work plus both the producer and consumer dependent work

Exploiting the independent work

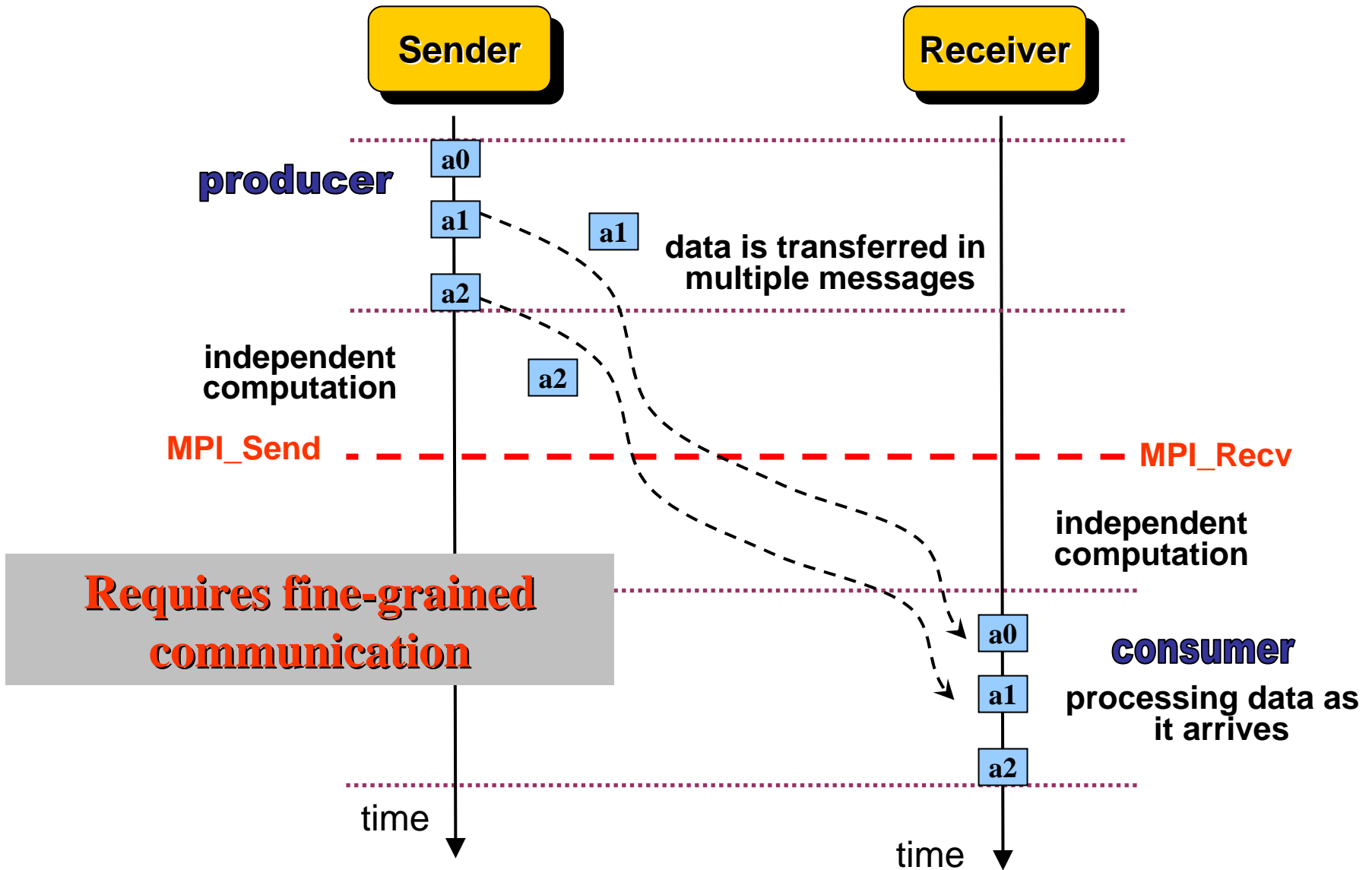
Effective case





Requires a mechanism to determine the arrival of data

Exploiting the dependent work plus the dependent work on both the producer and consumer



■ Code re-arrangement

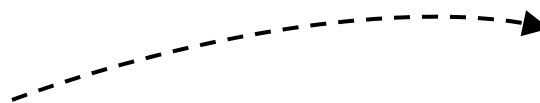
- ◆ Moving application code after the producer or before the consumer

■ Loop indexing

- ◆ Re-arranging loop indexes so that data which is not communicated is computed last on the producer or first on the consumer

■ Loop distribution

- ◆ Separates independent computation from the producer/consumer into multiple loops



```
computation 1
```

```
producer
```

MPI_Send



```
producer
```

MPI_Send

```
computation 1
```

We only time the computation time that would be available if these modifications are performed in the codes

| Application | Input | Number of processors |
|-------------|---------------------|----------------------|
| HYCOM | Large.inp | 1,006 |
| POP | x1 | 128 |
| Sweep3D | mk=1 320x320x400 | 1,024 |
| SAGE | timing_h | 1,024 |
| SAGE-AMR | timing_b | 1,024 |

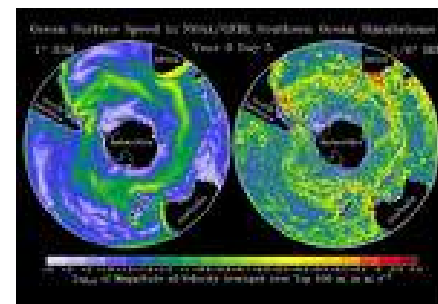
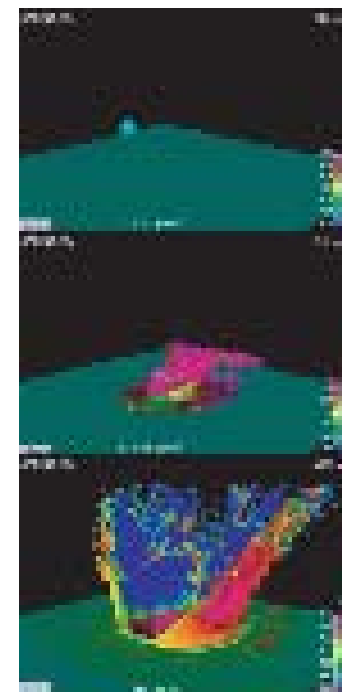


Image from ocean modeling



A sequence of images obtained from the execution of SAGE, an ASCII scientific application

Testbed cluster to timing computational work



APPRO cluster racks

4 TFLOPS cluster

256 dual-core AMD Opteron nodes: 1,024 cores

Processor speed 2 GHz

Memory 4GB/node

Voltaire 288-port InfiniBand 4X switch

Times are measured via hardware performance counters

MVAPICH on InfiniBand SDR 4X performance:

Latency: 4 microseconds

Bandwidth: 945MB/s

Modeling various network profiles to assess the sensitivity to low network performance

Model latency range: 1, 2, 4, and 8 microseconds

Model bandwidth range: 1 MB/s – 5GB/s

We are assuming zero overhead when communicating

Potential overlap on HYCOM

Measurements on independent work on the *barotropic*

| | ubavg | pbavg | vbavg |
|-------------------------|--------------|------------|--------------|
| Independent computation | | | |
| producer | 0 | 576 | 282 |
| consumer | 0 | 0 | 0 |
| Loop distribution | 192 | 192 | 0 |
| Loop indexing | | | |
| producer | 0 | 27.7 | 12.2 |
| consumer | 37.3 | 37.3 | 37.3 |
| Code re-arrangement | 0 | 0 | 0 |
| Total | 229.3 | 833 | 331.5 |

Times in usec

2.32

POP 2.78

SAGE 6.25

SAGE-AMR 2.84

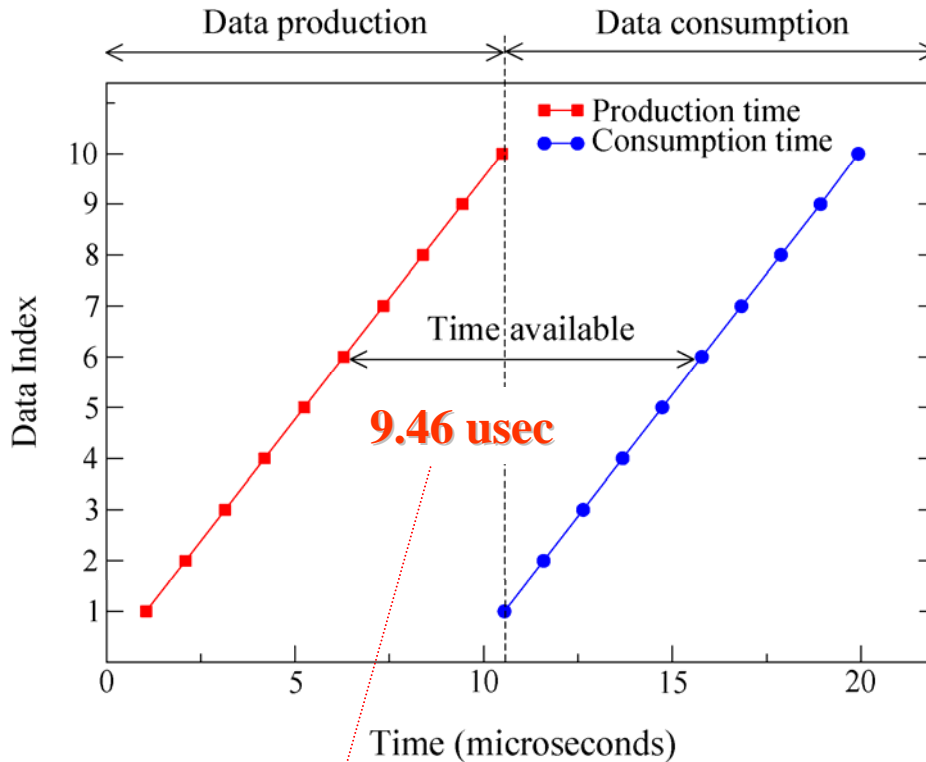
larger than the communication costs on InfiniBand network

Potential overlap on Sweep3D

Data structures exchanged: *phiib* and *phijb*

phijb data updates every 45ns with no independent work available

Measurements on dependent work for the *phiib*

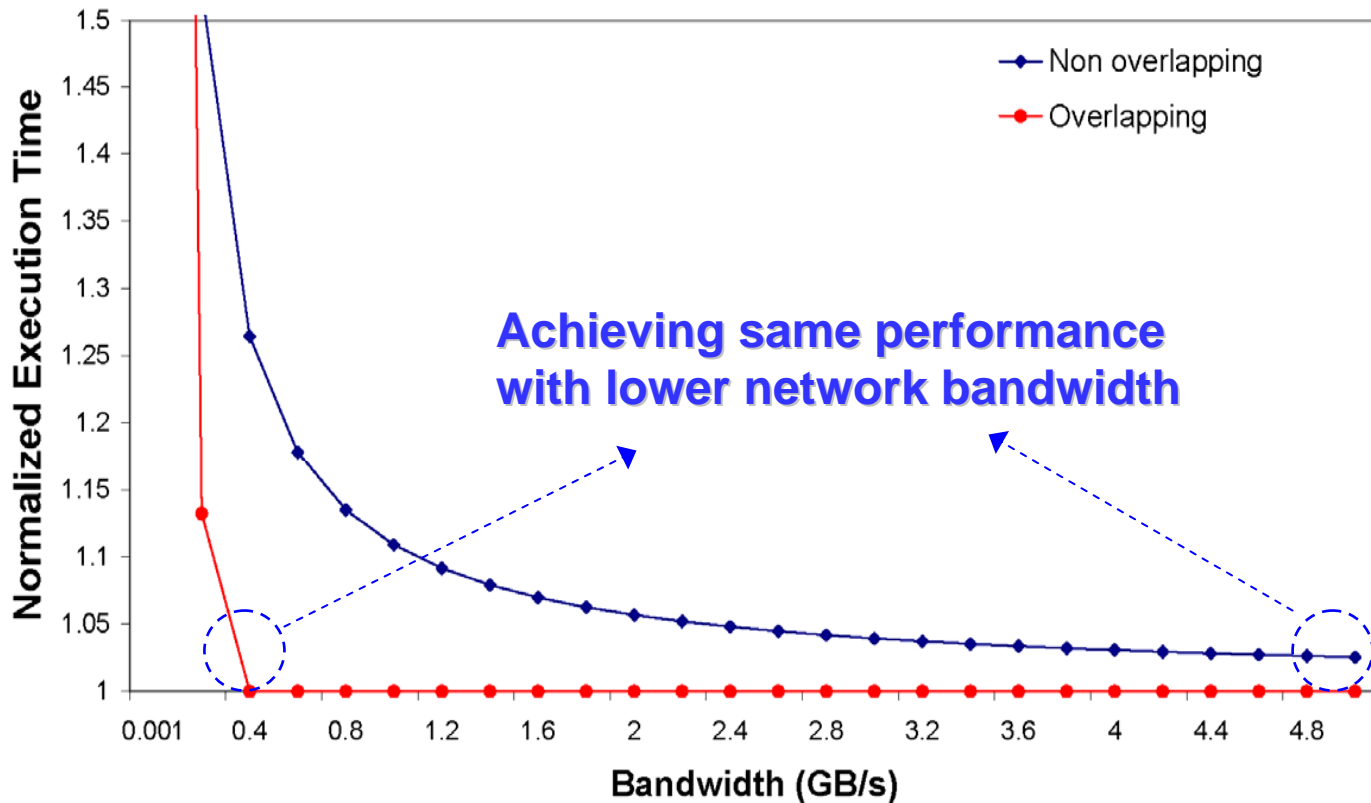


2.31

larger than the communication costs on InfiniBand

Application sensitive to network bandwidth

HYCOM



Performance on the *barotropic*

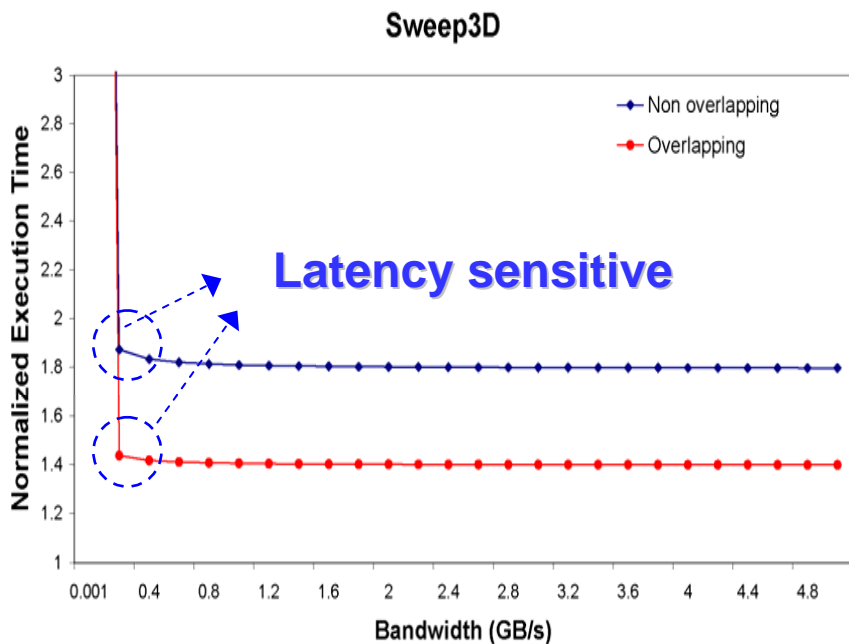
PAL Tolerating high network latency on Sweep3D

Application sensitive to network latency

Normalized execution time

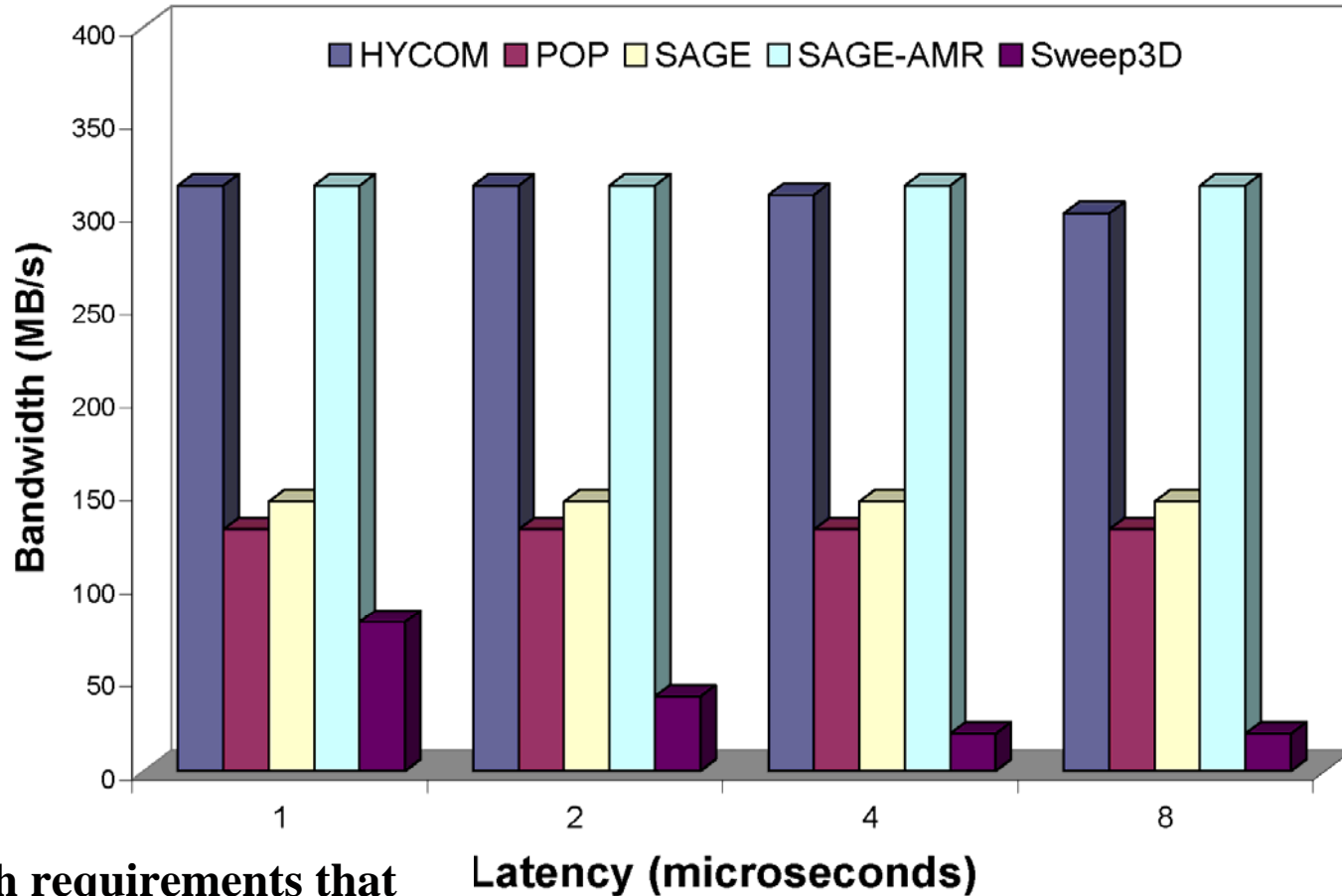
| latency | non over. | over. |
|---------|-----------|-------|
| 1 usec | 1.20 | 1.10 |
| 2 usec | 1.39 | 1.20 |
| 4 usec | 1.79 | 1.39 |
| 8 usec | 2.59 | 1.79 |

The same performance on a higher network latency



The *phiib* communication is limiting the performance

Bandwidth requirements when overlapping



Bandwidth requirements that achieves the same performance as non-overlapping on a network bandwidth of 5GB/s

- **Developed an analytical method to analyze the potential communication/computation overlap**
- **Distinguished two sources of overlap: independent and dependent**
- **The overlap can be exploited on networks with InfiniBand's latency and bandwidth**
- **Exploiting overlap reduces applications' network requirements**